



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

| APPLICATION NO.  | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|--|-------------|----------------------|---------------------|------------------|
| 10/522,897   | 02/01/2005  | Marc Vertes          | FR920050802US1      | 8427             |
| 35525 7590 02/19/2009<br>IBM CORP (YA)<br>C/O YEE & ASSOCIATES PC<br>P.O. BOX 802333<br>DALLAS, TX 75380 |             |                      |                     |                  |
| EXAMINER   |             |                      |                     |                  |
| RIAD, AMINE  |             |                      |                     |                  |
| ART UNIT   |             | PAPER NUMBER         |                     |                  |
| 2113   |             |                      |                     |                  |
| NOTIFICATION DATE  |             | DELIVERY MODE        |                     |                  |
| 02/19/2009   |             | ELECTRONIC           |                     |                  |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ptonotifs@yeciipaw.com

### Office Action Summary

**Application No.**

10/522,897

**Applicant(s)**

VERTES ET AL.

**Examiner**

AMINE RIAD

**Art Unit**

2113

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 13 November 2008.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 40-58 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 40-58 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/CDC)
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date: \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_
- Paper No(s)/Mail Date: \_\_\_\_\_

**Detailed Action**

Claims 40-58 have been presented for examination.

Claims 40-58 have been rejected.

Examiner notes here that after an updated search, the Examiner made the decision to withdraw the allowed claims, in view of the new art applied in this Office Action.

***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 40-58 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kelkar U.S. Patent 7,058,846 in view of Franckowiak U.S. Patent Application 2003000356.

In regard to claims 40, 58 .  
computer implemented method for replicating a software application,  
the computer implemented method comprising:  
executing the software application on a primary node to form a master application;  
identifying resources and dependencies required by the master application to form  
required resources; (Column 4; lines 49-53)  
Kelkar does not disclose *updating the required resources dynamically on the primary  
node; generating a structure of the master application and a dynamic graph of the  
required resources from the required resources;*

*replicating the resources by transferring the structure to a set of secondary nodes via a network to form a replica; wherein the set of secondary nodes comprises one or more secondary nodes; restoring the replica on the set of secondary nodes to form a set of clone software applications, wherein the set of clone software applications comprises one or more clone software applications; executing the set of clone software applications on the set of secondary nodes, without loss of context; and updating the set of clone software applications with incremental updates of the required resources of the master application to create a hot standby application.*

Franckoniak teaches updating the required resources dynamically on the primary node; (Paragraph 42) generating a structure of the master application and a dynamic graph of the required resources from the required resources;(Figure 2 ; item 56) replicating the resources by transferring the structure to a set of secondary nodes via a network to form a replica; wherein the set of secondary nodes comprises one or more secondary nodes; restoring the replica on the set of secondary nodes to form a set of clone software applications, wherein the set of clone software applications comprises one or more clone software applications; executing the set of clone software applications on the set of secondary nodes, without loss of context; and updating the set of clone software applications with incremental updates of the required resources of the master application to create a hot standby application.(Figure 2; items 58,60,68,70,72,74) [This figure clearly shows that whatever process takes place in the first node the system will mirror that in the second node]

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate *updating the required resources dynamically on the primary node; generating a structure of the master application and a dynamic graph of the required resources from the required resources; replicating the resources by transferring the structure to a set of secondary nodes via a network to form a replica; wherein the set of secondary nodes comprises one or more secondary nodes; restoring the replica on the set of secondary nodes to form a set of clone software applications, wherein the set of clone software applications comprises one or more clone software applications; executing the set of clone software applications on the set of secondary nodes, without loss of context; and updating the set of clone software applications with incremental updates of the required resources of the master application to create a hot standby application of Franckowiak into the method for replicating a software application of Kelkar*. A person of ordinary skill in the art would have been motivated to apply *updating the required resources dynamically on the primary node; generating a structure of the master application and a dynamic graph of the required resources from the required resources; replicating the resources by transferring the structure to a set of secondary nodes via a network to form a replica; wherein the set of secondary nodes comprises one or more secondary nodes; restoring the replica on the set of secondary nodes to form a set of clone software applications, wherein the set of clone software applications comprises one or more clone software applications; executing the set of clone software applications on the set of secondary nodes, without loss of context; and updating the set of clone software applications with*

*incremental updates of the required resources of the master application to create a hot standby application of Franckowiak* because as Franckowiak discloses "In an effort to achieve high availability, RCS application processors are paired to form mated processor pairs in an active/standby"

Arrangement. When a fault occurs on the active processor, the standby process is elevated to the active role to continue providing service.

In regard to claim 41

Kelkar discloses the computer implemented method according to claim 40, wherein replicating the resources further comprises:

creating and maintaining a dependency tree, based on the dynamic graph, supplying, at all times, information on the replicated resources. (Column 4; lines 53-56) [the synchronization of resource configuration necessitates respecting a dependency hierarchy, and a dynamic information provision]

In regard to claim 42

Franckowiak disclose the computer implemented method according to claim 40, wherein replicating the resources further comprises:

checkpointing the resources on the set of secondary nodes, wherein the checkpointing having an adjustable period. (Paragraph 74;" DML also performs a periodic heartbeat, sending information from the active application to the standby side so that the standby side can detect when it has missed a physical push or has lost communication with the active side.)

In regard to claim 43

Kelkar discloses the computer implemented method according to claim 42, wherein replicating the resources further comprises:  
capturing the resources on the primary node to create captured required resources;  
transferring the captured required resources over the network to the set of secondary nodes; and restoring the captured required resources on the set of secondary nodes.(Column 3; lines 33-39)

In regard to claim 44

Kelkar discloses the computer implemented method according to claim 42, wherein replicating the resources further comprises:  
optimizing the checkpointing. (Column 3; lines 37-39)

In regard to claim 45

Kelkar discloses the computer implemented method according to claim 44, wherein the checkpointing is incremental. (Column 5; lines 34-35)

In regard to claim 46

Kelkar disclose the computer implemented method according to claim 44, wherein the checkpointing is discriminating.(Column 5; lines 34-35)

In regard to claim 47

Kelkar discloses the computer implemented method according to claim 42, wherein the checkpointing further comprises at least one of the following: processing a synchronization barrier (Figure 1; item 1.2 and 1.4 updating of storage resource synchronization barrier and update completed) and (Figure 2; items 215A and 215B)

; managing resources (Figure 3; item 360A); managing system resources (Figure 1; item 104A); and managing process resources (Figure 2; item 110A).

In regard to claim 48

Kelkar discloses the computer implemented method according to claim 40, wherein replicating the resources further comprises:

replicating applicative data files between the primary node, whereon the software application is run, and a stand-by node. (Column 5; lines 8-14)

In regard to claim 49

Kelkar discloses the computer implemented method according to claim 40, wherein replicating the resources further comprises:

ensuring functional continuity of the software application in a multi-computer architecture cluster, the software application being executed at a given time on one of the computers of the cluster, called the primary node, while other computers of the cluster are called a set of secondary nodes, wherein ensuring functional continuity further comprises: replicating the software application on at least one of the secondary nodes to provide a set of clones of the application, wherein the set of clones comprises one or more clones; updating the set of clones, and responsive to detecting an event affecting the primary node, switching from the software application being executed on the primary node, to the software application being executed on the set of clones.

(Figure 2; [this figure shows 2 nodes 110A and 110B])

In regard to claim 50

Kelkar discloses the computer implemented method according to claim 49, wherein



replicating the software application is of a holistic nature. (Column 3; lines 36-39 "the fact that the copying is synchronized makes the replica consistent")

In regard to claim 51

Kelkar discloses the computer implemented method according to claim 49, wherein updating the set of clones further comprises updating the set of clones of the application. (Column 3; lines 33-36)

In regard to claim 52

Kelkar discloses the computer implemented method according to claim 49, wherein ensuring functional continuity further comprises supervising a state of the resources necessary to operate the software application. (Column 8; lines 48-50)

In regard to claim 53

Kelkar discloses the computer implemented method according to claim 49, wherein detecting an event affecting the primary node further comprises:

responsive to detecting an event affecting the primary node, electing a clone to be substituted for the primary node of the software application, wherein the secondary node on which the clone elect is installed becomes a new primary node. (Column 9; lines 14-19) [Examiner understands that when concurrency happens the manager is forced to elect, and that how one clone gets to be chosen over the other one]

In regard to claim 54

Kelkar discloses the computer implemented method according to claim 53, wherein replicating the resources further comprises:

recording, on the set of clones, messages received by the primary node, the messages

being injected into the clone elected as the new primary node when switching.(Column 7; lines 5-8) and (Column 6; lines 1-2)

In regard to claim 55

Kelkar discloses the computer implemented method according to claim 40, wherein replicating the resources further comprises:

optimization of information processing resources by load sharing and dynamic process distribution. (Column 1; line 28)

In regard to claim 56

Kelkar discloses the computer implemented method according to claim 40, wherein replicating the resources further comprises:

performing non-interruptive maintenance by process relocation upon request, over a data- processing resource network. (Summary) [Examiner considers maintenance a deliberate failure HINT shutting down purposely]

In regard to claim 57

Kelkar discloses the computer implemented method according to claim 40, wherein replicating the resources further comprises:

preserving applicative context in a mobile application. (Column 10; line 55)

### **Response to Applicant Argument**

Applicant's argument submitted 11/13/2008 has been fully considered and is not persuasive in view of the new reference Franckowiak United States Patent Application Publication 2003/0005356.

### **Contact**

Any inquiry concerning this communication or earlier communications from the examiner should be directed to AMINE RIAD whose telephone number is (571)272-8185. The examiner can normally be reached on 8-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert Beausoliel can be reached on 571-272-3645. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

**AR**

**/Robert W. Beausoliel, Jr./**  
**Supervisory Patent Examiner, Art Unit 2113**